

**LANGUAGE INPUT ARCHITECTURE FOR CONVERTING ONE TEXT FORM TO
ANOTHER TEXT FORM WITH TOLERANCE TO SPELLING, TYPOGRAPHICAL,
AND CONVERSION ERRORS**

Publication number: JP2003514304T

Publication date: 2003-04-15

Inventor:

Applicant:

Classification:

- International: G06F17/21; G06F17/22; G06F17/27; G06F17/28;
G06F17/21; G06F17/22; G06F17/27; G06F17/28;
(IPC1-7); G06F17/21; G06F17/27

- European: G06F17/27C; G06F17/22E2; G06F17/27A4;
G06F17/28D2; G06F17/28K

Application number: JP20010536716T 20001013

Priority number(s): US19990163902P 19991105; US20000606660
20000628; WO2000US28486 20001013

Also published as:

WO0135250 (A3)
WO0135250 (A2)
US7302640 (B2)
US6848080 (B1)
US2005086590 (A1)
US2005044495 (A1)
CN1387650 (A)
CN1205572C (C)

less <<

Report a data error here

Abstract not available for JP2003514304T

Abstract of corresponding document: WO0135250

A language input architecture converts input strings of phonetic text (e.g., Chinese Pinyin) to an output string of language text (e.g., Chinese Hanz) in a manner that minimizes typographical errors and conversion errors that occur during conversion from the phonetic text to the language text. The language input architecture has a search engine, one or more typing models, a language model, and one or more lexicons for different languages. Each typing model is trained on real data, and learns probabilities of typing errors. The typing model is configured to generate a list of probable typing candidates that may be substituted for the input string based on probabilities of how likely each of the candidate strings was incorrectly entered as the input string. The probable typing candidates may be stored in a database. The language model provides probable conversion strings for each of the typing candidates based on probabilities of how likely a probable conversion output string represents the candidate string. The search engine combines the probabilities of the typing and language models to find the most probable conversion string that represents a converted form of the input string. By generating typing candidates and then using the associated conversion strings to replace the input string, the architecture eliminates many common typographical errors. When multiple typing models are employed, the architecture can automatically distinguish among multiple languages without requiring mode switching for entry of the different languages.

Data supplied from the esp@cerenet database - Worldwide

(19) 日本国特許庁 (J P)

(12) 公表特許公報 (A)

(11) 特許出願公表番号

特表2003-514304

(P2003-514304A)

(43) 公表日 平成15年4月15日 (2003.4.15)

(51) Int.Cl. ⁷	識別記号	F I	テーマコード (参考)
G 0 6 F 17/21	5 9 2	G 0 6 F 17/21	5 9 2 F 5 B 0 0 9
17/27		17/27	5 9 2 J 5 B 0 9 1
			X

審査請求 未請求 予備審査請求 有 (全 51 頁)

(21) 出願番号 特願2001-536716(P2001-536716)
 (86) (22) 出願日 平成12年10月13日 (2000. 10. 13)
 (85) 翻訳文提出日 平成14年5月2日 (2002. 5. 2)
 (86) 国際出願番号 P C T / U S 0 0 / 2 8 4 8 6
 (87) 国際公開番号 W O 0 1 / 0 3 5 2 5 0
 (87) 国際公開日 平成13年5月17日 (2001. 5. 17)
 (31) 優先権主張番号 6 0 / 1 6 3 , 9 0 2
 (32) 優先日 平成11年11月5日 (1999. 11. 5)
 (33) 優先権主張国 米国 (US)
 (31) 優先権主張番号 0 9 / 6 0 6 , 6 6 0
 (32) 優先日 平成12年6月28日 (2000. 6. 28)
 (33) 優先権主張国 米国 (US)

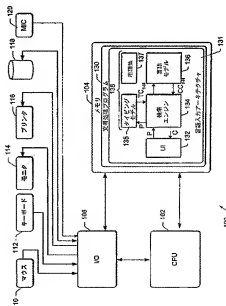
(71) 出願人 マイクロソフト コーポレーション
 MICROSOFT CORPORATION
 アメリカ合衆国 ワシントン州 98052-
 6399 レッドモンド ワン マイクロソフ
 ト ウェイ (番地なし)
 (72) 発明者 カイーフ リー
 アメリカ合衆国 98072 ワシントン州
 ウッディンビル 214 ウェイ ノースイ
 ースト 14233
 (74) 代理人 弁理士 谷 義一 (外2名)

最終頁に続く

(54) 【発明の名称】 スペルミス、タイプミス、および変換誤りに耐性のある、あるテキスト形式から別のテキスト形式に変換する言語入力アーキテクチャ

(57) 【要約】

言語入力アーキテクチャにより、表音テキスト (例えば、中国語のピンイン) の入力文字列を言語テキスト (例えば、漢字) の出力文字列に変換するが、表音テキストから言語テキストに変換する際に発生するタイプミスおよび変換誤りを極力減らす。この言語入力アーキテクチャは、検索エンジン、1つまたは複数のタイピングモデル、言語モデル、および異なる言語用の1つまたは複数の用語集を備える。各タイピングモデルは、実際のデータをもとにトレーニングされ、タイピング誤りの確率を学習する。タイピングモデルは、入力文字列の代わりにできる有望なタイピング候補のリストを生成するように構成される。この言語モデルは、前記タイピング候補のそれぞれについて有望な変換文字列を供給する。検索エンジンは、タイピングモデルと言語モデルの確率を組み合わせて、入力文字列の変換された形式を表す最も有望な変換文字列を見つける。



【特許請求の範囲】

【請求項1】 スペルミスを含む入力文字列を受け取る工程と、
少なくとも一部は統計的言語モデルを使用して前記スペルミスを修正する工程と
と
を具えたことを特徴とする方法。

【請求項2】 前記修正する工程は、
N重文字統計的言語モデルを使用することを含むことを特徴とする請求項1記載の方法。

【請求項3】 タイピングモデルを使用して前記スペルミスを修正するため
可能な候補文字列を生成する工程
をさらに具えたことを特徴とする請求項1記載の方法。

【請求項4】 前記統計的言語モデルと前記タイピングモデルから返された
組み合わせ確率により複数の可能な候補文字列を分析する工程
をさらに具えたことを特徴とする請求項3記載の方法。

【請求項5】 入力文字列を受け取る工程と、
前記候補文字列が前記入力文字列として間違って入力された確率に基づいて前
記入力文字列を置き換えるのに使用できる、少なくとも1つの候補文字列を判別
する工程と、
前記候補文字列を使用して少なくとも1つの出力文字列を導出する工程と、
前記入力文字列を前記出力文字列に変換する工程と
を具えたことを特徴とする方法。

【請求項6】 前記入力文字列は表音テキストを含み、前記出力文字列は言語
テキストを含むことを特徴とする請求項5記載の方法。

【請求項7】 前記入力文字列はピンインを含み、前記出力文字列は漢字を
含むことを特徴とする請求項5記載の方法。

【請求項8】 前記判別する工程は、
データベースから前記1つ又は複数の候補文字列を取得することを含むことを
特徴とする請求項5記載の方法。

【請求項9】 トレーニングテキストを入力した複数のユーザから集めたデ

一タをもとに、前記候補文字列が前記表音列として間違っ て入力された前記確率を導出する工程

をさらに具えたことを特徴とする請求項5記載の方法。

【請求項10】 前記判別する工程は、

複数の異なる方法で前記入力文字列をセグメント分割し、前記入力文字列を置き換えるのに使用できる複数の候補文字列を生成し、前記候補文字列のそれぞれが前記候補文字列を前記入力文字列として間違っ て入力した確率に基づくことを特徴とする請求項5記載の方法。

【請求項11】 前記使用は、

前記候補文字列のそれぞれを出力文字列と関連付けることを含むことを特徴とする請求項10記載の方法。

【請求項12】 前記入力文字列を置き換えるのに使用できる複数の候補文字列を判別する工程と、ここで、前記候補文字列のそれぞれは、前記候補文字列を前記入力文字列として間違っ て入力する確率に基づくものであり、

前記複数の候補文字列を使用して、複数の関連する出力文字列を導出する工程と、

前記確率に応じて前記候補文字列を選択し、前記変換のため前記選択した候補文字列と関連付けられる前記出力文字列を使用する工程とをさらに具えたことを特徴とする請求項5記載の方法。

【請求項13】 ユーザが入力している前記入力文字列と同じ行に前記出力文字列を表示する工程をさらに具えたことを特徴とする請求項5記載の方法。

【請求項14】 プロセッサ上で実行すると請求項5記載の前記方法を実行するようコンピュータに指令するコンピュータ実行可能命令を具えたことを特徴とする1つ又は複数のコンピュータ読み取り可能媒体。

【請求項15】 複数の異なる方法で入力文字列をセグメント分割し、前記入力文字列を置き換えるのに使用できる複数の候補文字列を生成する工程と、ここで、前記候補文字列のそれぞれは、前記候補文字列を前記入力文字列として間違っ て入力する確率に基づくものであり、

少なくとも1つの出力文字列を前記候補文字列のそれぞれと関連付ける工程とを具えたことを特徴とする方法。

【請求項16】 前記入力文字列は表音テキストを含み、前記出力文字列は言語テキストを含むことを特徴とする請求項15記載の方法。

【請求項17】 前記入力文字列はピンインを含み、前記出力文字列は漢字を含むことを特徴とする請求項15記載の方法。

【請求項18】 前記入力文字列はピンインと英語の組み合わせを含み、前記出力文字列は漢字と英語の組み合わせを含むことを特徴とする請求項15記載の方法。

【請求項19】 最高の確率が設定された特定の候補文字列を選択し、表音テキストの前記入力文字列を前記特定の候補文字列と関連する前記出力文字列に変換する工程

をさらに具えたことを特徴とする請求項15記載の方法。

【請求項20】 プロセッサ上で実行すると請求項15記載の前記方法を実行するようコンピュータに指令するコンピュータ実行可能命令を具えたことを特徴とする1つ又は複数のコンピュータ読み取り可能媒体。

【請求項21】 入力文字列を受け取る工程と、

少なくとも1つのトレーニングテキストを入力する複数のユーザから集めた実際のデータをもとにトレーニングされたタイピングモデルを使用して、前記入力文字列を可能な修正について評価する工程を具えたことを特徴とする方法。

【請求項22】 言語モデルを使用し、前記入力文字列の言語の文脈に基づいて前記入力文字列を置き換えるため有望な候補文字列を導出する工程をさらに具えたことを特徴とする請求項21記載の方法。

【請求項23】 タイピングモデルを構築する工程と、

前記タイピングモデルをトレーニングして、ユーザが第2の文字列が入力されたときに第1の文字列を入力するつもりであった確率を求める工程とを具え、ここで、前記トレーニングは、少なくとも1つのトレーニングテキストを入力する複数のユーザから集めたデータに基づくことを特徴とする方法。

【請求項24】 前記トレーニングは、発音が等しいすべての文字列を個々の音節に対応付けることを含むことを特徴とする請求項23記載の方法。

【請求項25】 前記トレーニングは、
複数の文字からなる文字列を読み取る工程と、
音節を前記文字列内の対応する文字に対応付ける工程と、
個々の音節について、前記音節に対応付ける前記文字列内の前記文字の頻度カウント数を維持する工程と、
前記音節が前記頻度カウント数に基づいて前記文字列の正しい入力を表す確率を決定する工程と
を含むことを特徴とする請求項23記載の方法。

【請求項26】 プロセッサ上で実行すると請求項23記載の前記方法を実行するようコンピュータに指令するコンピュータ実行可能命令を具えたことを特徴とする1つ又は複数のコンピュータ読み取り可能媒体。

【請求項27】 タイピングモデルをトレーニングする方法であって、
複数の文字からなるテキスト文字列を読み取る工程と、
音節を前記テキスト文字列内の対応する文字に対応付ける工程と、
個々の音節について、前記音節に対応付ける前記テキスト文字列内の前記文字の頻度カウント数を維持する工程と、
前記音節が前記頻度カウント数に基づいて前記テキスト文字列の正しい入力を表す確率を決定する工程と
を具えたことを特徴とする方法。

【請求項28】 前記テキスト文字列は、表音テキストを含むことを特徴とする請求項27記載の方法。

【請求項29】 前記テキスト文字列は、表音テキストと非表音テキストとの混合を含むことを特徴とする請求項27記載の方法。

【請求項30】 プロセッサ上で実行すると請求項27記載の前記方法を実行するようコンピュータに指令するコンピュータ実行可能命令を具えたことを特徴とする1つ又は複数のコンピュータ読み取り可能媒体。

【請求項31】 言語入力アーキテクチャであって、

入力文字列を受け取り前記入力文字列にスペルミスが含まれるユーザインタフェースと、

先行する文字列の文脈において前記入力文字列を評価し、前記入力文字列を置換して前記スペルミスを修正することができる有望な代替文字列を生成する言語モデルと

を具えたことを特徴とする言語入力アーキテクチャ。

【請求項32】 前記言語モデルは、N重文字統計的言語モデルを含むことを特徴とする請求項31記載の言語入力アーキテクチャ。

【請求項33】 前記候補文字列のそれぞれが前記入力文字列として間違っ
て入力されたタイピング誤りの確率に基づいて前記入力文字列に代用可能な、有望なタイピング候補のリストを生成するタイピングモデル

をさらに具えたことを特徴とする請求項31記載の言語入力アーキテクチャ。

【請求項34】 言語入力アーキテクチャであって、

入力文字列を受け取り前記入力文字列にスペルミスが含まれるユーザインタフェースと、

前記候補文字列のそれぞれが入力文字列として間違っ
て入力されたタイピング誤りの確率に基づいて前記入力文字列を置き換えられる有望なタイピング候補のリストを生成するタイピングモデルと

を具え、

ここで、前記タイピングモデルは少なくとも1つのトレーニングテキストを入力する複数のユーザから集めた実際にデータをもとにトレーニングされることを特徴とする言語入力アーキテクチャ。

【請求項35】 前記タイピングモデルは、第1の言語を使用してトレーニングされ、

前記候補文字列のそれぞれが前記入力文字列として間違っ
て入力されたタイピング誤りの確率に基づいて前記入力文字列に代用可能な、有望なタイピング候補のリストを生成する第2のタイピングモデル

をさらに具え、

前記第2のタイピングモデルは、第2の言語でトレーニングされることを特徴

とする請求項34記載の言語入力アーキテクチャ。

【請求項36】 言語入力アーキテクチャであって、

前記候補文字列のそれぞれが前記入力文字列として間違っ
て入力されたタイピング誤りの確率に基づいて表音テキストで書かれた入力文字列の代わりにできる有望なタイピング候補のリストを生成するタイピングモデルと、

前記タイピング候補のそれぞれについて言語テキストで書かれた出力文字列を供給する言語モデルと

を具えたことを特徴とする言語入力アーキテクチャ。

【請求項37】 前記表音テキストはピンインであり、前記言語テキストは漢字であることを特徴とする請求項36記載の言語入力アーキテクチャ。

【請求項38】 前記タイピングモデルは、トレーニングテキストを入力する複数のユーザから集めたデータを使用してトレーニングされることを特徴とする請求項36記載の言語入力アーキテクチャ。

【請求項39】 表音テキストで書かれた前記入力文字列を受け取るユーザインタフェースをさらに具えたことを特徴とする請求項36記載の言語入力アーキテクチャ。

【請求項40】 前記タイピング候補を格納するデータベースをさらに具えたことを特徴とする請求項36記載の言語入力アーキテクチャ。

【請求項41】 請求項36記載の前記言語入力アーキテクチャを具えたコンピュータ読み取り可能媒体上に実現されることを特徴とするワードプロセッサ。

【請求項42】 言語入力アーキテクチャであって、

入力文字列を受け取り、候補文字列が間違っ
て前記入力文字列として入力されたタイピング誤りの確率を決定するタイピングモデルと、

出力文字列が前記候補文字列を表す場合の言語テキストの確率を決定する言語モデルと、

前記タイピング誤りの確率と前記言語テキストの確率に基づいて前記入力文字列を前記出力文字列に選択的に変換する検索エンジンと

を具えたことを特徴とする言語入力アーキテクチャ。

【請求項43】 前記入力文字列は発音テキストを含み、前記出力文字列は言語テキストを含むことを特徴とする請求項42記載の方法。

【請求項44】 前記入力文字列はピンインを含み、前記出力文字列は漢字を含むことを特徴とする請求項42記載の方法。

【請求項45】 前記入力文字列は発音テキストと非発音テキストの組み合わせを含み、前記出力文字列は言語テキストと非発音テキストの組み合わせを含むことを特徴とする請求項42記載の方法。

【請求項46】 前記タイピングモデルは、トレーニングテキストを入力する複数のユーザから集めたデータを使用してトレーニングされることを特徴とする請求項42記載の言語入力アーキテクチャ。

【請求項47】 前記入力文字列を受け取り、前記出力文字列を表示するユーザインタフェースをさらに具えたことを特徴とする請求項42記載の言語入力アーキテクチャ。

【請求項48】 前記タイピング候補を格納するデータベースをさらに具えたことを特徴とする請求項42記載の言語入力アーキテクチャ。

【請求項49】 前記入力文字列も含むセンテンス内のすでに入力されているテキストをタイピングモデルに供給するセンテンス文脈モデルをさらに具え、ここで、前記タイピングモデルは、前記入力文字列と前記センテンス内のテキストの組み合わせを使用して、前記タイピング誤りの確率を導出するように構成されていることを特徴とする請求項42記載の言語入力アーキテクチャ。

【請求項50】 請求項42記載の前記言語入力アーキテクチャを具えたコンピュータ読み取り可能媒体上に実現されることを特徴とするワードプロセッサ。

【請求項51】 コンピュータで実行可能な命令を有する1つ又は複数のコンピュータ読み取り可能媒体であって、プロセッサで実行されたときコンピュータに、

スペルミスを含む入力文字列を分析させ、

統計的言語モデルを使用して前記スペルミスを修正することを命令させることを特徴とするコンピュータ読み取り可能媒体。

【請求項52】 コンピュータで実行可能な命令を格納した1つ又は複数のコンピュータ読み取り可能媒体であって、プロセッサで実行されたときコンピュータに、

入力文字列を受け取らせ、

少なくとも1つのトレーニングテキストを入力する複数のユーザから集めた実際のデータをもとにトレーニングされたタイピングモデルを使用して前記入力文字列を可能な修正について評価させることを特徴とするコンピュータ読み取り可能媒体。

【請求項53】 コンピュータで実行可能な命令を格納した1つ又は複数のコンピュータ読み取り可能媒体であって、プロセッサで実行されたときコンピュータに、

入力文字列を受け取らせ、

前記候補文字列が前記入力文字列として間違って入力された確率に基づいて前記入力文字列を置き換えるのに使用可能な、少なくとも1つの候補文字列を判別させ、

前記候補文字列を使用して少なくとも1つの出力文字列を導出させ、

前記入力文字列を前記出力文字列に変換させることを特徴とするコンピュータ読み取り可能媒体。

【発明の詳細な説明】

【0001】

(技術分野)

本発明は、言語入力方法およびシステムに関する。より詳細には、本発明は、テキスト入力時に発生するタイプミスおよびある言語形式から別の言語形式への変換時に発生する変換誤りの両方に対する誤り耐性 (error tolerance) のある言語入力方法およびシステムを実現する。

【0002】

(発明の背景)

言語固有のワードプロセッサソフトが何年もの間存在している。より複雑なワードプロセッサは、スペルおよび文法の訂正など、ユーザに高等なツールを提供し、文書を起草する際に補助する。例えば、多数のワードプロセッサは、スペルミスの単語や文法的に間違っているセンテンス構造を識別し、場合によっては、識別された誤りを自動的に訂正することができる。

【0003】

一般に、誤りがテキストに入り込む原因は2つある。原因の1つは、ユーザが単に、正しいスペルまたはセンテンス構造を知らないことである。ワードプロセッサは、ユーザが正しいスペルまたは言葉使いを選択する際に手助けするために提案することができる。第2のより一般的な誤りの原因は、ユーザが、正しいスペルまたは文法構造を知っていても単語またはセンテンスを間違ってコンピュータに入力することである。このような状況では、ワードプロセッサは、不適切に入力された文字列を識別し、意図した単語または語句に訂正するのに大変役立つことが多い。

【0004】

入力誤りは、欧文文字を使用しない言語用に設計されたワードプロセッサで起こりがちである。英語バージョンのQWERTYキーボードなど言語固有のキーボードは、多くの言語の場合、そのような言語がキーボードのキーとして使いやすく配列できるよりも多くの文字を有するため、存在していない。例えば、多くのアジア言語は数千文字を含む。これだけ多くの様々な文字について別々のキー

をサポートするキーボードを構築することは実際上不可能である。

【0005】

費用のかかる言語および方言固有のキーボードを設計するのではなく、言語固有の文書処理システムは、ユーザが小さな文字セットキーボード（例えば、QWERTYキーボード）から表音テキスト（phonetic text）を入力し、その表音テキストを言語テキストに変換することができる。「表音テキスト」は、所与の言語を発話したときに発生する音を表すが、「言語テキスト」はテキストで表示されたとき実際に書かれた文字を表す。例えば、中国語では、ピンインは表音テキストの一例であり、漢字は言語テキストの一例である。表音テキストを言語テキストに変換することにより、多くの様々な言語を、従来のコンピュータおよび標準QWERTYキーボードを使用して言語固有のワードプロセッサによって処理できる。

【0006】

表音入力が必要とするワードプロセッサでは、2種類の入力誤りが発生する可能性がある。1つの種類の誤りは、一般的なタイプミスである。しかし、テキストにタイプミスがないとしても、文書処理エンジンが表音テキストを意図しない文字テキストに誤って変換するという誤りもありえる。これら2つの問題の両方が、同じ表音テキスト入力文字列で作用した場合、一連の複数の誤りが生じることがある。状況によっては、語句またはセンテンスの文脈全体を時間をかけて調査しなければ、タイピングで生じる誤りが容易に追跡できない場合がある。

【0007】

本明細書で説明する本発明は、表音テキストをタイプした場合のユーザによる前者のタイプの入力誤りを主に対象としているが、文書処理エンジンで生じた変換誤りに対する耐性も提供する。このようなタイピング誤りと関連する問題をよく説明するために、表音テキストであるピンインを言語テキストである漢字に変換する中国語ベースのワードプロセッサを考察する。

【0008】

表音テキストを入力すると多くの場合タイピング誤りが増える理由はいくつかある。理由の1つは、英語用のキーボードでの平均タイピング精度は英語を話す

国々の場合よりも中国の場合の方が低いということである。第2の理由は、表音はそれほど頻繁に使用されているわけではないということである。初期の教育期間中、ユーザは、例えば、英語を話すユーザが英語で単語を綴ることを教わるほど、表音綴りを勉強、学習する傾向はない。

【0009】

表音テキスト入力中のタイピング誤りが増える第3の理由は、多くの人々が標準語ではなく方言を母語として話すというものである。表音テキストの起源である標準語は、第2言語である。特定の方言およびアクセントでは、話し言葉は、対応する適切な表音と一致しない場合があり、従って、ユーザが表音テキストをタイプすることをより難しくする。例えば、多くの中国人はさまざまな中国語方言を自分の第1言語として話しており、ピンインの起源である北京語を第2言語として教えられている。例えば、いくつかの中国語方言では、一定の文脈で「h」と「w」を発音する際に区別がなく、他の方言では、「ng」と「n」についても同じことが言え、さらに他の方言では、「r」は明瞭に発音されない。そのため、北京語を第2言語として話す中国人ユーザは、ピンインを入力しようとしてタイピング誤りをしがちである。

【0010】

タイピング誤りが増える理由としては他に、表音テキストをタイプしながら誤りをチェックすることが困難であるということが考えられる。これは、1つには、表音テキストが長く判読しにくくなり、読むことが困難になる傾向があるということによる。見たものがタイプしたものである英語ベースのテキスト入力と対照的に、表音テキストの入力は、「見たものが得られるもの」でないことが多い。むしろ、ワードプロセッサは、表音テキストを言語テキストに変換する。そのため、ユーザは一般に、表音テキストに誤りがないか調べることをしないが、表音テキストが言語テキストに変換されるまで多少待つ。

【0011】

この最後の理由により、タイピング誤りは、ピンイン入力の文脈ではきわめてうっとうしいものである。ピンイン文字列は、文字間にスペーシングがないため再検討、訂正が非常に難しい。その代わりに、ピンイン文字は、ピンイン文字で

構成される単語の数に関係なく混在する。さらに、ピンイン→漢字変換はすぐに実行されることはないが、追加ピンインを入力するにつれ正しい解釈が定式化され続ける。従って、ユーザが間違ったピンイン記号をタイプした場合、単一の誤りが変換プロセスによって複合され、下流に伝播して、いくつかの追加誤りの原因となる場合がある。その結果、システムが漢字に確定変換しその後ユーザが誤りのあったことに気づくまでに、1回訂正するのにユーザは数回バックスペースを入力せざるをえないため、誤り訂正の時間が長くなる。システムによっては、元の誤りを明らかにできない場合さえある。

【0012】

表音入力中に間違いが頻繁に生じることが予想されるため、表音入力の誤りを許容できるシステムが必要である。システムは、表音列がわずかに誤りのある文字列を含むとしても正しい答えを返すのが望ましい。

【0013】

言語固有ワードプロセッサは他の問題に直面しており、これは、入力問題とは別であって、異なる言語から同じテキストに単語を入力するために2つの言語間でモードを切り替えることに関するものである。例えば、技術用語（例えば、Internet）などの英単語と翻訳が難しい用語（例えば、頭字語、記号、サネーム、社名など）を含む文書草稿を中国語で作成することがよくある。従来のワードプロセッサでは、ユーザは異なる単語を入力するときに、一方の言語を他方の言語にモード切替をする必要がある。そのため、ユーザが異なる言語から単語を入力したい場合、ユーザはテキスト入力についての思考を停止し、モードのある言語から別の言語に切り替えて、単語を入力し、それからモードを最初の言語に戻す必要がある。このため、ユーザのタイピング速度が著しく低下し、ユーザは自分の注意をテキスト入力作業と、言語モードの変更という異質な制御作業との間で切り替える必要がある。

【0014】

従って、モード切替を必要としない「モードレス」システムが必要なのである。モードを避けるために、システムは、タイプしている言語を検出し、文字シーケンスを1単語ずつ一方の言語または他方の言語に動的に変換することができな

ければならない。

【0015】

しかし、これは、両方の文脈で多くの文字列が適切な場合があるため、思ったほど簡単ではない。例えば、多くの有効な英単語はさらに有効なピンイン文字列でもある。さらに、ピンイン入力時に中国語文字間、および中国語文字と英単語との間にスペースがないため、曖昧さが増すことがある。

【0016】

例えば、ユーザがピンイン入力テキスト「woshiyigezhongguoren」をタイプすると、システムはこの文字列を中国語文字「 」（一般に「私は中国人である」と翻訳される）に変換する。

【0017】

ときには、「woshiyigezhongguoren」とタイプする代わりに、ユーザは次のようにタイプします。

wosiyigezhongguoren（誤りは「sh」と「s」の混同）

woshiyigezongguoren（誤りは「zh」と「z」の混同）

woshiygezhongguoren（誤りは、「y」の後の「i」の脱落）

woshiyigezhonggouren（誤りは、「ou」の並列）

woshiyigezhongguiren（誤りは「i」と「o」の混同）

【0018】

発明者は、中国語などの難しい外国語でスペルミスの修正を可能にする文書処理システムと方法をすでに開発しており、自動言語認識機能により複数言語のモードレス入力が可能になっている。

【0019】

（発明の概要）

言語入力アーキテクチャにより、表音テキスト（例えば、中国語のピンイン）の入力文字列を言語テキスト（例えば、漢字）の出力文字列に変換するが、表音テキストから言語テキストに変換する際に発生するタイプミスおよび変換誤りを極力減らす。言語入力アーキテクチャは、文書処理プログラム、電子メールプロ

グラム、表計算ソフト、ブラウザなどさまざまな分野で実装することができる。

【0020】

一実装では、言語入力アーキテクチャは入力文字列、記号、またはその他のテキスト要素を受け取るユーザインタフェースを備える。入力文字列は、表音テキストおよび非表音テキスト、さらに1つまたは複数の言語を含むことができる。このユーザインタフェースにより、ユーザは、異なるテキスト形式または異なる言語の入力のモードを切り替えることなく単一の編集行に入力テキスト文字列を入力することができる。この方法により言語入力アーキテクチャは、ユーザが使いやすいように複数の言語のモードレス入力を実現している。

【0021】

この言語入力アーキテクチャはさらに、検索エンジン、1つまたは複数のタイピングモデル、言語モデル、および異なる言語用の1つまたは複数の用語集を備える。検索エンジンは、ユーザインタフェースから入力文字列を受け取り、入力文字列を1つまたは複数のタイピングモデルに分配する。各入力モデルは、各候補文字列が入力文字列として間違っ て入力された場合のタイピング誤り確率に基づいて入力文字列に置換できる有望なタイピング候補のリストを生成するように構成されている。有望なタイピング候補は、データベースに格納できる。

【0022】

タイピングモデルは、トレーニングテキストを入力している多くのトレーナから集めたデータでトレーニングする。例えば、中国語の文脈では、トレーナはピンインで書かれているトレーニングテキストを入力する。トレーニングテキストを入力しているときに見つかった誤りを使用して、タイピング誤りを修正するのに使用できるタイピング候補と関連する確率を計算する。複数のタイピングモデルを採用している場合、各タイピングモデルを異なる言語でトレーニングできる。

【0023】

一実施形態では、タイピングモデルは、入力テキストの文字列を読み取り、音節に対応するタイプされた各文字列の文字に対応付けることによりトレーニングできる。タイプされた各文字が音節の1つに対応づけられる回数を表す頻度カウ

ントを保持し、各音節のタイピングの確率をその頻度カウントから計算する

タイピングモデルは、入力文字列内に存在する可能性のあるタイプミスの原因となりうるタイピング候補の集まりを返す。タイピング候補は、入力文字列と同じ言語またはテキスト形式で書かれる。

【0024】

検索エンジンはタイピング候補を言語モデルに渡し、これが各タイピング候補の可能性のある変換文字列となる。より具体的には、言語モデルは三重文字言語モデルであり、有望な変換出力文字列が前の2つのテキストエレメントに基づいて候補文字列を表す言語テキスト確率を求めようとする。変換文字列は、入力文字列と異なる言語または異なるテキスト形式で書かれている。例えば、入力文字列は、中国語ピンインまたはその他の表音テキストで構成され、出力文字列は漢字またはその他の言語テキストで構成される。

【0025】

タイピングおよび言語モデルで求めた確率に基づき、検索エンジンは最高の確率を示す関連するタイピング候補および変換候補を選択する。検索エンジンは、入力文字列（例えば、表音テキストで書かれている）を言語モデルから返された変換候補からなる出力文字列に変換し、入力されたテキスト形式（例えば、表音テキスト）を他のテキスト形式（例えば、言語テキスト）に置き換える。この方法により、表音テキストの入力時のユーザによる入力誤りがなくなる。

【0026】

複数言語を使用する場合、出力文字列は変換候補と入力文字列の一部分（変換なし）との組み合わせを持つことができる。後者の例では、中国語ベースの言語入力アーキテクチャは、両方の変換されたピンインー漢字テキストを変換されていない英語テキストとともに出力する。

【0027】

ユーザインタフェースは、入力文字列の入力にそのまま使用する同じ編集行に出力文字列を表示する。この方法では、変換は自動的に行われ、ユーザが追加テキストを入力すると同時に実行される。

【0028】

図全体を通して類似の構成要素および特徴を参照するのに同じ番号を使用している。

【0029】

(好ましい実施形態の詳細な説明)

本発明は、言語のある形式（例えば、表音バージョン）から言語の別の形式（例えば、書き言葉バージョン）に変換する言語入力システムおよび方法に関連する。このシステムおよび方法は、テキスト入力時に発生するスペルミスおよびタイプミスおよびある言語形式から別の言語形式への変換時に発生する変換誤りに対する誤り耐性がある。説明のため、本発明は、汎用コンピュータで実行される文書処理プログラムの一般的文脈で説明する。ただし、本発明は、文書処理以外の異なる多くの環境に実装することができ、またさまざまな種類のデバイスで実施することができる。他に、電子メールプログラム、表計算ソフト、ブラウザなどでの文脈が考えられる。

【0030】

言語入力システムは、統計的言語モデルを採用して非常に高い精度を達成している。一実施例では、言語入力アーキテクチャは、自動的な最高確率ベースの方法で統計的言語モデリング (statistical language modeling) を使用し、単語をセグメント化し、語彙を選択し、トレーニングデータをフィルタ処理し、可能な最良の変換候補を求める。

【0031】

ただし、統計的センテンスベース言語モデリング (Statistical sentence-based language modeling) では、ユーザの入力が完全であると仮定している。実際には、ユーザの入力にはタイピング誤りやスペルミスが多数ある。従って、言語入力アーキテクチャは、確率論的スプリングモデルを使用して、ありがちなタイピング誤りやスペルミスを許容しながら正しいタイピングを受け入れる1つまたは複数のタイピングモデルを含む。タイピングモデルを英語や中国語など複数言語についてトレーニングし、どれくらいの確からしさで入力シーケンスがある言語の単語であって別の言語の単語ではないかを識別するようにできる。両方のモデルは並列実行でき、その言語

モデル（例えば、中国語モデル）により誘導されて最も可能性の高い文字シーケンス（つまり、英語および中国語の文字）を出力する。

【0032】

<コンピュータシステム実施例>

図1は、中央処理装置（CPU）102、メモリ104、および入出力（I/O）インタフェース106を備えるコンピュータシステム実施例100を示している。CPU 102は、メモリ104およびI/Oインタフェース106と通信する。メモリ 104は、揮発性メモリ（例えば、RAM）および不揮発性メモリ（例えば、ROM、ハードディスクなど）を表す。

【0033】

コンピュータシステム100は、I/Oインタフェース106を介して接続された1つまたは複数の周辺装置を備える。周辺装置実施例は、マウス110、キーボード112（例えば、英数字QWERTYキーボード、表音キーボードなど）、ディスプレイモニタ114、プリンタ116、周辺記憶装置118、およびマイクロホン120を備える。例えば、コンピュータシステムは、汎用コンピュータで実装できる。従って、コンピュータシステム100は、メモリ104に格納され、CPU 102で実行されるコンピュータのオペレーティングシステム（図に示されていない）を実装する。オペレーティングシステムは、ウィンドウ操作環境をサポートするマルチタスクオペレーティングシステムであるのが好ましい。適当なオペレーティングシステムの例として、Microsoft Corporation社のWindows（登録商標）ブランドのオペレーティングシステムがある。

【0034】

ハンドヘルドデバイス、マルチプロセッサシステム、マイクロプロセッサベースのまたはプログラム可能な家電製品、ネットワークPC、ミニコンピュータ、メインフレームコンピュータなど、他のコンピュータシステム構成を使用できることに注意されたい。さらに、図1にはスタンドアローンのコンピュータが示されているが、通信ネットワーク（例えば、LAN、インターネットなど）を介してリンクされているリモート処理デバイスによってタスクが実行される分散コン

ビューティング環境で言語入力システムを実用することもできる。分散コンピュータ環境では、プログラムモジュールをローカルとリモートの両方のメモリ記憶デバイスに配置できる。

【0035】

データまたは文書処理プログラム130は、メモリ104に格納され、CPU102によって実行される。他のプログラム、データ、ファイルなども、メモリ104に格納できるが、説明を簡単にするため示していない。文書処理プログラム130は、表音テキストを受け取り、自動的に言語テキストに変換するように構成されている。より具体的には、文書処理プログラム130は、説明のためメモリ内に格納されプロセッサによって実行されるコンピュータソフトウェアとして実装されている言語入力アーキテクチャ131を実装する。文書処理プログラム130は、アーキテクチャ131に加えて他のコンポーネントも備えることができるが、そのようなコンポーネントは、文書処理プログラムに標準のものであると考えられるため、図に詳細に示したり、詳細に説明していない。

【0036】

文書処理プログラム130の言語入力アーキテクチャ131は、ユーザインタフェース(UI)132、検索エンジン134、1つまたは複数のタイピングモデル135、言語モデル136、および異なる言語用の1つまたは複数の用語集137を備える。アーキテクチャ131は、言語独立である。UI132および検索エンジン134は、汎用であり、どの言語でも使用できる。アーキテクチャ131は、言語モデル136、タイピングモデル135、用語集137を変更することにより特定の言語に合わせる。

【0037】

検索エンジン134および言語モジュール136はいっしょに使用することで、表音テキストー言語テキスト間のコンバータ138を形成する。タイピングモデル135の助けを借りて、コンバータ138はユーザのタイピング誤りおよびスペルミスに対し許容性を持つ。この開示の目的のために、「テキスト」は、1つまたは複数の文字および／または文字以外の記号とする。「表音テキスト」は一般に、所与の言語を話すときに生じる音声を表す英数字テキストである。「言

語テキスト」は、書き言葉を表す文字と非文字記号である。「非表音テキスト」は、所与の言語を話すときに生じる音声を表さない英数字テキストである。非表音テキストは、言語テキスト以外の書き言葉を表す句読点、特殊文字、および英数字テキストを含む場合がある。

【0038】

たぶん、より一般的に述べると、表音テキストは、書いたときに欧文文字ベースの文字セットを使用しない所与の言語を話したときに出る音声を表す欧文文字セット（例えば、英語のアルファベット）で表された英数字でよい。言語テキストは、所与の言語に対応する書かれた記号である。

【0039】

説明の目的のために、ワードプロセッサ130は、中国語ベースのワードプロセッサの文脈で説明し、言語入力アーキテクチャ131はピンインを漢字に変換するように構成されている。つまり、表音テキストはピンインであり、言語テキストは漢字である。しかし、言語入力アーキテクチャは、言語独立であり、他の言語にも使用できる。例えば、表音テキストは日本語の話し言葉の形態でよいが、言語テキストは漢字などの日本語の書き言葉を表す。アラビア語、韓国語、インド語、その他のアジア言語などを含むが、これだけに限定されない他の例も多数存在する。

【0040】

表音テキストは、マウス110、キーボード112、またはマイクロホン120などの1つまたは複数の周辺入力デバイスを介して入力する。この方法で、ユーザは、キー入力または音声による表音テキスト入力が可能である。音声入力の場合、コンピュータシステムはさらに、話し言葉を受け取る音声認識モジュール（図に示されていない）を実装し、表音テキストに変換することができる。以下の説明では、キーボード112によるテキストの入力をフルサイズの標準英数字QWERTYキーボードで実行すると想定している。

【0041】

UI 132では、表音テキストを入力と同時に表示する。このUIは、グラフィカルユーザインタフェースであるのが好ましい。UI 132の詳細な説明

は、引用により本発明に取り込まれている、「LANGUAGE INPUT USER INTERFACE」という表題の同時係属出願第_____号にある。

【0042】

ユーザインタフェース132は、表音テキスト(P)を検索エンジン134に渡し、さらに、これは表音テキストをタイピングモデル137に渡し、タイピングモデル137は、表音テキストに誤りが含まれていると思われる場合にユーザが意図した表音テキストの適当な編集結果と考えられるさまざまなタイピング候補(TC₁, . . . , TC_N)を生成する。タイピングモデル137は、妥当な確率が設定されている複数のタイピング候補を検索エンジン134に渡し、さらに、これはタイピング候補を言語モデル136に渡し、この言語モデル136は、処理中のセンテンスの文脈の範囲内でタイピング候補を評価し、ユーザが意図した表音テキストの変換された形式を表すと考えられる言語テキストで書かれているさまざまな変換候補(CC₁, . . . , CC_N)を生成する。変換候補は、タイピング候補と関連付けられている。

【0043】

表音テキストから言語テキストへの変換は1対1変換ではない。同じあるいは類似した表音テキストが言語テキスト内の多数の文字または記号を表すことがある。従って、表音テキストの文脈は、言語テキストへの変換前に解釈される。他方、非表音テキストの変換は、通常、表示される英数字テキストが英数字入力と同じである直接的な1対1変換となる。

【0044】

変換候補(CC₁, . . . CC_N)が検索エンジン134に戻され、このエンジンにより、タイピング候補および変換候補のうちどれがユーザが意図するものである確率が最も高いかを判別する統計分析が実行される。確率が計算されると、検索エンジン134により、確率が最も高い候補が選択され、変換候補の言語テキストがUI 132に返される。UI 132では、表音テキストを変換候補の言語テキストに置き換え、同じ行に表示する。一方で、新規入力した表音テキストが新規挿入された言語テキストの前の行に表示され続ける。

【0045】

ユーザが言語テキストを検索エンジン134で選択したもののから変更したい場合、ユーザインタフェース132に、選択が実際に意図した回答である可能性の高さの順序でランク付けられた他の高確率の候補の第1のリストが表示される。ユーザがまだ可能な候補に満足しない場合、UI 132は可能なすべての選択肢を与える第2のリストを表示する。第2のリストは、確率またはその他の測定基準（例えば、ストロークカウントまたは中国語文字の複雑さ）に関してランク付けることができる。

【0046】

<言語入力アーキテクチャ>

図2は、言語入力アーキテクチャ131の詳細を示している。アーキテクチャ131は、タイプミスおよび変換誤りの両方を含む、言語入力の誤り耐性をサポートする。UI 132、検索エンジン134、言語モデル136、およびタイピングモデル135に加えて、アーキテクチャ131はさらに、エディタ204およびセンテンス文脈モデル216を備える。センテンス文脈モデル216は、検索エンジン134に結合されている。

【0047】

ユーザインタフェース132は、表音テキスト（例えば、中国語ピンインテキスト）および非表音テキスト（例えば、英語）などの入力テキストを1つまたは複数の周辺装置（例えば、キーボード、マウス、マイクロホン）から受け取り、その入力テキストをエディタ204に渡す。エディタ204は、検索エンジン132がタイピングモデル135および言語モデル136とともに入力テキストを、言語テキスト（例えば、漢字テキスト）などの出力テキストに変換するように要求する。エディタ204は、出力テキストをUI 132に戻して表示させる。

【0048】

検索エンジン134は、ユーザインタフェース132から入力テキスト文字列を受け取ると、その入力テキスト文字列をタイピングモデル135の1つまたは複数の送りに、さらにセンテンス文脈モデル216にも送る。タイピングモデル135は、入力テキスト内のタイピング誤りのアプリアリな確率を測定する。タイピングモデル135は、入力誤り（例えば、タイプミス）を修正することを実際

に求めるユーザによって入力された入力テキストの有望なタイピング候補を生成して出力する。一実施形態では、タイピングモデル135は候補データベース210内で有望な候補を検索する。他の実装では、タイピングモデル135は統計ベースのモデリングを使用して、入力テキストの有望な候補を生成する。

【0049】

センテンス文脈モデル216は、オプションで、タイピングモデル135で使用する検索エンジン132にセンテンス内のすでに入力されているテキストを送ることができる。この方法では、タイピングモデルは、テキストの新しい文字列とセンテンスにすでに入力されているテキストの文字列との組み合わせに基づいて有望なタイピング候補を生成することができる。

【0050】

「タイピング誤り」、「タイプミス」、および「スペルミス」という用語は、相互に入れ替えることができ、入力テキストのキー入力時に生じた誤りを指すことは明白である。音声入力の場合、このような誤りは、音声入力の不適切な認識から生じることがある。

【0051】

タイピングモデル135は、有望なタイピング候補をすべて返すか、または確率の低い有望なタイピング候補を取り除き、それにより、高い確率の有望なタイピング候補のみを検索エンジン134に返すことができる。さらに、タイピングモデル135よりはむしろ検索エンジン134が取り除き機能を実行できることも明白であろう。

【0052】

本発明の一態様によれば、タイピングモデル135は、ありがちなタイプミスを観察するため数百あるいは数千のトレーニングにセンテンス入力を依頼して集めた実際のデータ212を使用してトレーニングされる。以下では「タイピングモデルのトレーニング」という見出しのもとでタイピングモデルおよびトレーニングについて詳述する。

【0053】

検索エンジン134は、タイピングモデル135から返された有望なタイピン

グ候補のリストを言語モデル136に送る。簡単にいうと、言語モデルは、語句やセンテンスなど、指定された文脈内に単語またはテキスト文字列がある確率を測定する。つまり、言語モデルは、項目（単語、文字、英字など）のシーケンスを取り、そのシーケンスの確率を推定することができる。言語モデル136は、検索エンジン134から有望なタイピング候補と前のテキストとを組み合わせ、タイピング候補に対応する言語テキストの1つまたは複数の候補を生成する。

【0054】

コーパスデータまたはその他の種類のデータ214を使用して、三重文字言語モデル136をトレーニングする。トレーニングコーパス214は、新聞記事などの日々のテキストなどの一般的な任意の種類のデータ、あるいは特定の分野（例えば、医薬品）を対象とするテキストなどの環境固有のデータとすることができる。言語モデル136のトレーニングは、文書処理技術の分野では知られており、ここでは詳述しない。

【0055】

言語入力アーキテクチャ131は、入力テキスト文字列の入力時に生じる誤りを許容し、入力文字列となる単語およびセンテンスで最も確率の高いものを返そうとする。言語モデル136は、タイピングモデル135でユーザが入力した入力文字列に対しどのセンテンスが最も妥当かを判別する際に役立つ。2つのモデルは、入力された文字列 s が辞書、つまり $P(w | s)$ から認識可能で有効な単語 w である確率として統計的に記述することができる。ベイズの公式を使用すると、確率 $P(w | s)$ は次のように記述される。

【0056】

【数1】

$$P(w|s) = \frac{P(s|w) \cdot P(w)}{P(s)}$$

【0057】

分母 $P(s)$ は、入力文字列が与えられたときに可能な意図した単語を比較する目的では同じである。した h がって、分析は、分子の積 $P(s | w) \cdot P(w)$ のみに関係し、確率 $P(s | w)$ はスペルまたはタイピングモデルを表し、確

率 $P(w)$ は言語モデルを表す。より具体的には、タイピングモデル $P(s|w)$ は、 X を入力するつもりの人が代わりに Y を入力することになる確率を記述するが、言語モデル $P(w)$ はセンテンス文脈を与えたときに特定の単語が生成されているべき確率を記述する。

【0058】

ピンインを漢字に変換する文脈では、確率 $P(w|s)$ は $P(H|P)$ と言い換えることができ、 H は漢字文字列、 P はピンイン文字列を表す。目標は、 $P(H|P)$ を最大にする、最も確率の高い中国語文字 H' を見つけることである。従って、確率 $P(H|P)$ は、入力されたピンイン文字列 P が有効な漢字文字列 H である確率である。 P は固定されており、従って $P(P)$ は与えられたピンイン文字列に対し一定であるため、ベイズ公式により、次のように確率 $P(H|P)$ が小さくなる。

【0059】

$$\begin{aligned} H' &= \arg \max_H P(H|P) \\ &= \arg \max_H P(P|H) * P(H) \end{aligned}$$

【0060】

確率 $P(P|H)$ はスペルまたはタイピングモデルを表す。通常、漢字文字列 H は、さらに、複数の単語 $W_1, W_2, W_3, \dots, W_n$ に分割され、確率 $P(P|H)$ は次のように推定できる。

【0061】

$$Pr(P|H) \propto P(P_{(w)} | W_i)$$

【0062】

$P_{(w)}$ は、単語 W_i に対応するピンイン文字列のシーケンスである。

【0063】

従来技術の統計ベースのピンイン-漢字変換システムでは、確率 $P(P_{(w)} | W_i)$ は、 $P_{(w)}$ が単語 W_i の受け入れ可能なスペルである場合に1に設定され、 $P_{(w)}$ が単語 W_i の受け入れ可能なスペルでない場合に0に設定される。そのため、従来のシステムは、誤って入力された文字に対する耐性がない。一部のシステムでは、「南部訛りの発音」機能があり、このような問題に対応しているが、こ

れはプリセット値確率1および0を採用している。さらに、このようなシステムは、データ駆動方式でないためタイピング誤りのうちごくわずかししか扱えない（実際のタイピング誤りから学習する）。

【0064】

対照的に、本発明で説明している言語入力アーキテクチャでは、タイピングモデルと言語モデルの両方を利用して変換を実行する。タイピングモデルでは、実際のコーパスから確率 $P(P_{\text{in}} | W_i)$ をトレーニングすることにより誤って入力した文字に対する誤り耐性が可能になっている。タイピングモデルを構築する方法は多数ある。理論的には、すべての可能な $P(P_{\text{in}} | W_i)$ をトレーニングできるが、実際には、パラメータが多すぎる。トレーニングする必要のあるパラメータの個数を減らす1つの方法として、1文字単語のみを考察し、発音が等価なすべての文字を単一の音節に対応づける方法がある。中国語にはおおよそ406個の音節があり、これは本質的に $P(\text{ピンインテキスト} | \text{音節})$ をトレーニングし、各文字を対応する音節に対応づけることである。以下では「タイピングモデルのトレーニング」という見出しのもとでこれについて詳述する。

【0065】

言語入力アーキテクチャ131では、広範な確率が計算される。ピンイン→漢字変換の目標は、確率 $P(P | H)$ を最大にする漢字文字列 H をつけることである。これは、最大の確率を最良の漢字シーケンスとして求める W_i を選択することにより実行する。実際、よく知られているViterbi Beam検索のような効率的な検索方法を使用できる。Viterbi Beam検索法の詳細については、「Automatic Speech Recognition」(Kluwer Academic Publishers, 1989) という表題の Kai-Fu Lee の記事、および「Automatic Speech and Speaker Recognition - Advanced Topics」(Kluwer Academic Publishers, 1996) という表題の Chin-Hui Lee, Frank K. Soong, Kuldeep K. Paliwal の記事を記載されている。

【0066】

確率 $P(H)$ は、言語モデルを表し、所与の単語列のアプリオリな確率を測定する。統計的言語モデル構築の一般的な方法として、プレフィックスツリー風のデータ構造を利用して、知られているテキストのトレーニングセットから N 重文字言語モデルを構築する方法がある。広く使用されている統計的言語モデルの実施例として、 N 重文字マルコフモデルがあり、これについては、Frederick Jelinek 著「Statistical Methods for Speech Recognition」(The MIT Press, Cambridge, Massachusetts, 1997) に説明がある。プレフィックスツリーデータ構造 (a. k. a. サフィックスツリー、または PAT ツリー) の使用により、高レベルアプリケーションで言語モデルを素早く横断し、上述のように、実質的にリアルタイムに実行する特性を持つ。 N 重文字言語モデルでは、テキスト全体を通して文字列 (サイズ N) 内の特定のアイテム (単語、文字など) の出現数をカウントする。このカウントを使用して、そのアイテムの列の使用の確率を計算する。

【0067】

言語モデル 136 は、三重文字言語モデル (つまり、 $N=3$ とする N 重文字) であるのが好ましいが、二重文字が文脈によっては適当な場合がある。三重文字言語モデルは、英語に適しており、また中国にも十分機能するが、大きなトレーニングコーパスを利用すると想定している。

【0068】

三重文字モデルでは、次のように、次の文字を予測するためにテキスト文字列内の最も前の 2 つの文字を考慮する。

(a) 文字 (C) は、定義済み用語集を使用して離散言語テキストまたは単語 (W) にセグメント化され、ツリー内の各 W は 1 つまたは複数の C に対応づけられる。

(b) 前の 2 つの単語から単語のシーケンス (W_1, W_2, \dots, W_n) の確率を予測する。

【0069】

$P(W_1, W_2, W_3, \dots, W_n) \mid \square P(W_n \sim W_{n-1}, W_{n-2})$

【0070】

ただし、P O は言語テキストの確率を表す。

【0071】

W_n は、現在の単語である

W_{n-1} は、前の単語である

W_{n-2} は、 W_{n-1} の前の単語である

【0072】

図3は、ユーザによって入力された入力テキスト300の例を示しており、タイピングモデル135および言語モデル136に渡される。入力テキスト300を受け取ると、タイピングモデル135は入力テキスト300を異なる方法でセグメント化し、キーボード入力時に生じる可能性のあるタイプミスを考慮した有望なタイピング候補のリストを生成する。タイピング候補302は、各時間フレーム内に異なるセグメントがあり、前の単語の終了時間が現在の単語の開始時間となる。例えば、候補302の上行は、入力文字列300「ma f a n g n i t r y y i s . . .」を「ma」、「fan」、「ni」、「try」、「yi」などにセグメント分割する。タイピング候補302の第2行は、入力文字列「ma f a n g n i t r y y i s . . .」を異なる形で「ma」、「fang」、「nit」、「yu」、「xia」などにセグメント分割する。

【0073】

これらの候補は、データベースまたはその他の何らかのアクセス可能なメモリに格納できる。図3は、単なる一例にすぎず、入力テキストに対する有望なタイピング候補は多数あり得ることは明白であろう。

【0074】

言語モデル136は、センテンスの文脈で有望なタイピング候補302の各セグメントを評価し、関連する言語テキストを生成する。説明のため、有望なタイピングテキスト302の各セグメントおよび対応する有望な言語テキストはボックスにまとめられている。

【0075】

これらの候補から、検索エンジン134は、候補のうちどれがユーザが意図するものである確率が最も高いかを判別する統計分析を実行する。各行内のタイピング候補は、互いに関連がなく、検索エンジンは任意の行からさまざまなセグメントを自由に選択し、受け入れ可能な変換候補を定義することができる。図3の例では、検索エンジンは、ハイライト表示になっているタイピング候補304、306、308、310、312、および314が最高の確率であることを示していると判断している。これらの候補は、左から右に連結され、候補304の後に候補306が続くなどして、入力テキスト300の受け入れ可能な解釈を形成することができる。

【0076】

確率を計算した後、検索エンジン134は、確率が最高の候補を選択する。検索エンジンは次に、入力された表音テキストを選択した候補と関連する言語テキストに変換する。例えば、検索エンジンは入力テキスト300をボックス304、306、308、310、312、および314で示されている言語テキストに変換し、エディタ204を介して言語テキストをユーザインタフェース132に返す。句読点がユーザインタフェースに届いた後、つまり新しい入力テキスト文字列が新しいセンテンス内に入ると、タイピングモデル135は新しいセンテンス内の新しいテキスト文字列に対する操作を開始する。

【0077】

<一般的な変換>

図4は、表音テキスト（例えば、ピンイン）を言語テキスト（例えば、漢字）に変換する一般のプロセス400を示している。このプロセスは、言語入力アーキテクチャ131によって実装されており、図2をさらに参照して説明する。

【0078】

ステップ402では、ユーザインタフェース132は、ユーザが入力したピンインなどの表音テキスト列を受け取る。入力テキスト文字列には、1つまたは複数のタイプミスが含まれる。UI 132は、エディタ204を介して入力テキストを検索エンジン134に渡し、検索エンジンは入力テキストをタイピングモデル135とセンテンス文脈モデル216に分配する。

【0079】

ステップ404では、タイピングモデル135は入力テキストに基づいて有望なタイピング候補を生成する。候補を導く一方法として、入力テキスト文字列を異なるパーティションに分割し、その入力文字列セグメントに最もよく類似する候補をデータベース内で検索する。例えば、図3で、候補302は可能なセグメント「ma」、「fan」などを示すセグメンテーションを持つ。

【0080】

有望なタイピング候補が、検索エンジン134に返され、その後、言語モデル136に伝達される。言語モデル136は、有望なタイピング候補と前のテキストとを組み合わせ、タイピング候補に対応する言語テキストの1つまたは複数の候補を生成する。例えば、図3の候補302を参照すると、言語モデルはボックス302a-j内に言語テキストを可能な出力テキストとして返す。

【0081】

ステップ406では、検索エンジン134は、候補のうちどれがユーザが意図するものである確率が最も高いかを判別する統計分析を実行する。表音テキストに最も可能性の高いタイピング候補を選択した後、検索エンジンは入力された表音テキストをタイピング候補と関連する言語テキストに変換する。この方法により、表音テキストの入力時のユーザによる入力誤りがなくなる。検索エンジン134は、エディタ204を介して誤りのない言語テキストをUI 132に返す。ステップ408では、変換された言語テキストは、ユーザが表音テキストを入力し続けているUI 132の画面上の同じ行内位置に表示される。

【0082】

<タイピングモデルのトレーニング>

上で指摘したように、タイピングモデル135は、確率 $P(s|w)$ に基づいている。タイピングモデルでは、入力テキストを出力テキストに変換するのに使用できる異なるタイピング候補の確率を計算し、有望な候補を選択する。この方法で、タイピングモデルは、タイピング誤りが存在していても入力テキストの有望なタイピング候補を返すことにより誤りを許容する。

【0083】

本発明の一態様は、実際のデータからのタイピングモデル $P(s|w)$ のトレーニングに関するものである。タイピングモデルは、数百または好ましくは数千などできる限り多くのトレーナによるテキスト入力に基づいて開発またはトレーニングされる。トレーナは同じまたは異なるトレーニングデータを入力し、入力されたデータとトレーニングデータとの分散をタイピング誤りとして捕らえる。目標は、同じトレーニングテキストをタイプさせ、タイピングでの誤りの個数またはタイピング候補に基づいて確率を求めることである。このようにして、タイピングモデルはトレーナのタイピング誤りの確率を学習する。

【0084】

図5は、プロセッサ502、揮発性メモリ504、および不揮発性メモリ506を備えるトレーニング用コンピュータ500を示している。トレーニング用コンピュータ500では、ユーザが入力したデータ510から確率 $P(s|w)$ (つまり、 $P(s|w)$) を求めるトレーニングプログラム508を実行する。トレーニングプログラム508は、プロセッサ502で実行するように図に示されているが、不揮発性メモリ506のストレージからプロセッサにロードされる。トレーニング用コンピュータ500は、オンザフライでの入力時に、あるいは収集しメモリに格納した後にデータ510に基づいてトレーニングを行うように構成することができる。

【0085】

説明のため、中国語用に手直したタイピングモデルを考察し、中国語ピンインテキストは中国語文字テキストに変換する。この場合、数千人の人々にピンインテキストを入力してくれるよう勧誘する。好ましくは、数千個またはそれ以上のセンテンスを各人から収集し、目標はタイピングでの誤りの種類および個数が類似するようにすることである。タイピングモデルは、検索エンジンからピンインテキストを受け取り、入力文字列内の文字の置き換えに使用できる有望な候補を供給するように構成されている。

【0086】

タイピングモデル135をトレーニングするためにさまざまな手法を使用できる。一方法では、タイピングモデルは、単一文字テキストを考察し、すべての同

等な発音の文字テキストを単一音節に対応付けることにより直接トレーニングされる。例えば、中国語ピンインには400を超える音節がある。音節を与える表音テキストの確率(例えば、P(ピンインテキスト|音節))をトレーニングし、各文字テキストに対応する音節に対応付ける。

【0087】

図6は、音節対応付けトレーニング手法600を示している。ステップ602で、トレーニングプログラム508は、トレーナが入力したテキスト文字列を読みとる。テキスト文字列は、センテンスでも、また単語および/または文字のその他のグループでもよい。プログラム508は、音節をテキストの文字列内の対応する英字に合わせるか、または対応付ける(ステップ604)。各テキスト文字列では、各音節に対応付けた英字の頻度が更新される(ステップ606)。これは、ステップ608から「はい」分岐で表されているように、トレーナにより入力されたトレーニングデータに含まれるテキスト文字列ごとに繰り返される。最終的に、入力されたテキスト文字列は、中国語ピンインの多くのまたはすべての音節を表す。ステップ608から「いいえ」分岐により表されるように、すべての文字列が読み込まれたら、トレーニングプログラムは、各音節をタイプするユーザの確率P(ピンインテキスト|音節)を決定する(ステップ610)。一実装では、タイピングの確率は、最初にすべての音節を正規化することにより決定される。

【0088】

各音節は、隠マルコフモデル(HMM)として表すことができる。各入力キーは、HMMで対応付けられている状態のシーケンスとして表示できる。正しい入力および実際の入力をすりあわせて状態間の遷移確率を求める。異なるHMMを使用して、異なる技能レベルのタイピストをモデル化することができる。

【0089】

中国語で406個すべての音節をトレーニングするには、大量のデータが必要である。このデータ要件を緩和するために、異なる音節内の同じ文字を1つの状態として結びつける。これにより、状態の個数は27にまで減らされる(つまり、「a」から「z」までの26個の異なる英字に、不明な文字を表す文字1つ)

。このモデルは、三重文字言語モデルを利用するViterbi beam検索法に統合することもできる。

【0090】

さらに他のトレーニング手法では、トレーニングは英字の挿入（つまり、

【0091】

【数2】

$\emptyset \rightarrow x$

【0092】

）、英字の削除（つまり、

【0093】

【数3】

$x \rightarrow \emptyset$

【0094】

）、および一方の文字を他方に置換（つまり、

【0095】

【数4】

$x \rightarrow y$

【0096】

）などの単一文字編集の確率に基づく。このような単一文字編集の確率は次のように統計的に表すことができる。

【0097】

置換：P（xをyで置換）

挿入：P（xをyの前／後に挿入）

削除：P（xをyの前／後に削除）

【0098】

各確率（P）は、本質的に二重文字タイピングモデルであるが、隣接する文字

を超えるかなり広い文脈のテキストを考慮したN重文字タイピングモデルに拡張することもできる。従って、入力テキストの可能な文字列について、タイピングモデルは、まず正しい文字シーケンスを供給し、次に動的プログラミングを使用して正しい文字シーケンスを与えられた文字シーケンスに変換する最低コスト経路を求めることにより、すべての可能な文字シーケンスを生成する確率を持つ。コストを、最小数の誤り文字または他の何らかの測定基準として決めることができる。実際には、この誤りモデルは、Viterbi Beam検索法の一部として実装できる。

【0099】

タイピング誤りまたはスペルミス以外の種類の誤りは、本発明の範囲内でトレーニングできることは明白であろう。さらに、異なるトレーニング手法を使用し、本発明の範囲から逸脱することなくタイピングモデルをトレーニングすることができることも明白であろう。

【0100】

<モードレス入力の多言語トレーニング>

言語入力システムを悩ます他のやっかいな問題として、2つまたはそれ以上の言語を入力したときのモードの切り替えの必要性である。例えば、中国語でタイプしているユーザは、英語の単語を入力したい場合がある。従来の入力システムでは、ユーザは英単語のタイピングと中国語の単語のタイプのモード切替が必要である。残念なことに、ユーザが切り替えを忘れやすいということである。

【0101】

言語入力アーキテクチャ131(図1)をトレーニングして混合言語入力を受け入れトレーニングすることができ、従って、多言語文書処理システムにおいて2つまたはそれ以上の言語間のモード切替をなくすることができる。これは、「モードレス入力」と呼ばれる。

【0102】

この言語入力アーキテクチャは、中国語と英語を区別するなど、異なる言語の単語を自動的に識別するスペル/タイピングモデルを実装している。これは、多くの正当な英単語は正当なピンイン文字列であるため容易ではない。さらに、ピ

ンイン、英語、および中国語文字の間にスペースが入らないため、入力時に曖昧さが増すことがある。以下のベイズ規則を使用すると、

【0103】

$$\begin{aligned} H' &= \arg \max_{\mathbb{H}} P(H | P) \\ &= \arg \max_{\mathbb{H}} P(P | H) * P(H) \end{aligned}$$

【0104】

目的関数は、英語ではスペルモデル $P(P | H)$ 、中国語では言語モデル $P(H)$ の2つの部分があることで特徴付けることができる。

【0105】

混合言語入力を取り扱う一方法として、第2言語からの単語を第1言語の特殊カテゴリとして取り扱うことにより、第1言語（例えば、中国語）の言語モデルをトレーニングする方法がある。例えば、第2言語からの単語を第1言語の単一単語として取り扱う。

【0106】

例えば、中国語ベースの文書処理システムでは、英語キーボードを入力デバイスとして使用する。中国語ベースの文書処理システムで採用しているタイピングモデルは、中国語言語モデルであって、英単語と中国語単語を混ぜたテキストでトレーニングされる。

【0107】

混合言語入力を取り扱う第2の方法として、言語入力アーキテクチャで2つのタイピングモデル、中国語タイピングモデルと英語タイピングモデルを実装し、それぞれを別々にトレーニングする方法がある。つまり、中国語タイピングモデルは、上で説明した方法でトレーナにより入力された表音列などのキーボード入力のストリーム上でトレーニングされ、英語タイピングモデルは英語を話すトレーナによって入力された英語テキスト上でトレーニングされる。

【0108】

英語タイピングモデルは、以下の組み合わせとして実装することができる。

1. 中国語テキストに挿入された実際の英語上でトレーニングした一重文字言語モデル。このモデルは、多くの頻繁に使用される英単語を取り扱えるが、見た

ことのない英単語は予測できない。

【0109】

2. 3音節確率の英語スベルモデル。このモデルは、すべての3音節シーケンスに対し確率が0でないが、英語に似た単語になる可能性のある単語については確率が高くなる。これは、実際の英単語からもトレーニングでき、見たことのない英単語も取り扱える。

【0110】

これらの英語モデルは、一般に、英語テキストに対しては非常に高い確率、英語テキストのように見える英字列には高い確率、非英語テキストには低い確率を返す。

【0111】

図7は、図2のアーキテクチャ131から修正された言語入力アーキテクチャ700を示しており、これは複数のタイピングモデル135(1)～135(N)を採用している。各タイピングモデルは、特定の言語に合わせて構成されている。各タイピングモデル135は、単語と、特定の言語に共通する誤りを使用し、別々にトレーニングされる。従って、別々のトレーニングデータ212(1)～212(N)が、関連するタイピングモデル135(1)～135(N)について供給される。実施例では、英語に対して1つのタイピングモデル、中国語に対してもう1つのタイピングモデルというようにタイピングモデルを2つだけ使用している。ただし、言語入力アーキテクチャを修正して、2つよりも多いタイピングモデルを含めて、2つよりも多い言語の入力に対応するようにできることは明白であろう。また、言語入力アーキテクチャは、日本語、韓国語、フランス語、ドイツ語などの他の多くの多言語文書処理システムでも使用できることも指摘しておくべきであろう。

【0112】

言語入力アーキテクチャの操作時に、英語タイピングモデルは中国語タイピングモデルと並列に動作する。2つのタイピングモデルは互いに競合しており、入力したテキスト文字列が中国語文字列(誤りを含む)または英語文字列(さらに潜在的に誤りを含む)である可能性のある確率を計算することにより入力テキス

トが英語か中国語かを識別する。

【0113】

入力テキストの文字列またはシーケンスが明確に中国語ピンインテキストの場合、中国語タイピングモデルは英語タイピングモデルよりもかなり高い確率を返す。そこで、言語入力アーキテクチャは、入力されたピンインテキストを漢字テキストに変換する。入力テキストの文字列またはシーケンスが明確に英語（例えば、サーネーム、頭字語（「IEEE」）、会社名（「Microsoft」）、技術（「INTERNET」）、など）の場合、英語タイピングモデルは中国語タイピングモデルよりもかなり高い確率を示す。従って、このアーキテクチャは、英語タイピングモデルに基づいて入力テキストを英語テキストに変換する。

【0114】

入力テキストの文字列またはシーケンスが曖昧な場合、中国語および英語タイピングモデルは、中国語か英語かわからない曖昧さを解消するためにさらなる文脈から情報が得られるまで確率を計算し続ける。入力テキストの文字列またはシーケンスが中国語にも英語にも似ていない場合、中国語タイピングモデルは英語タイピングモデルよりも許容性が低い。そのため、英語タイピングモデルは確率が、中国語タイピングモデルよりも高くなる。

【0115】

多言語変換を説明するために、ユーザが「私はINTERNETマガジンを読むのが好きだ」という意味のテキスト文字列「woa idu internet z a z h i」を入力すると仮定する。初期文字列「woa idu」を受け取った後、中国語タイピングモデルは、英語タイピングモデルよりも高い確率となり、入力テキストのその部分を「INTERNET」に変換する。このアーキテクチャは、続いてタイプした曖昧な部分「i n t e r n e」を英字「t」がタイプされるまで探し続ける。このときに、英語タイピングモデルは、「INTERNET」について中国語タイピングモデルよりも高い確率を返し、言語入力アーキテクチャは入力テキストのこの部分を「INTERNET」に変換する。次に、中国語タイピングモデルは、「z a z h i」について英語タイピングモデルよりも高い確率を示し、言語入力アーキテクチャは入力テキストのその部分を「」

に変換する。

【0116】

<多言語入力変換>

図8は、タイプミスとともに入力された多言語入力テキスト文字列を誤りのない多言語出力テキスト文字列に変換するプロセス800を示している。このプロセスは、言語入力アーキテクチャ700によって実装されており、図7をさらに参照して説明する。

【0117】

ステップ802では、ユーザインタフェース132は、多言語入力テキスト文字列を受け取る。これは、少なくとも1つの他の言語（例えば、英語）の表音単語（例えば、ピンイン）および単語を含む。入力テキストはさらに、ユーザが表音単語および第2言語の単語を入力したときのタイプミスも含む場合がある。UI 132は、エディタ204を介して多言語入力テキスト文字列を検索エンジン134に渡し、検索エンジンは入力テキストをタイピングモデル135(1)～135(N)とセンテンス文脈モデル216に分配する。

【0118】

タイピングモデルはそれぞれ、ステップ804(1)～804(N)によって表されるような入力テキストに基づいて有望なタイピング候補を生成する。ステップ806では、妥当な確率が設定された有望なタイピング候補が検索エンジン134に返される。ステップ808では、検索エンジン134がタイピング確率とともにタイピング候補を言語モデル136に送る。ステップ810では、言語モデルが有望なタイピング候補と前のテキストとを組み合わせ、センテンスベースの文脈を用意し、図3に関して上で説明したように、そのタイピング候補を通じて経路を選択することによりタイピング候補に対応する言語テキストの1つまたは複数の変換候補を生成する。ステップ812では、検索エンジン134は統計分析を実行して、ユーザが意図する最高の確率を示す変換候補を選択する。

【0119】

ステップ814では、テキスト文字列に対する最も有望な変換候補が出力テキスト文字列に変換される。出力テキスト文字列は、言語テキスト（例えば、漢字

）と第2言語（例えば、英語）を含むが、タイピング誤りは省かれる。検索エンジン134は、エディタ204を介して誤りのない出力テキストをUI 132に返す。ステップ816では、変換された言語テキストは、ユーザが表音テキストを入力し続けているUI 132の画面上の同じ行内位置に表示される。

【0120】

上の例では、中国語が主言語であり、英語は第2言語である。2つの言語は両方とも、主言語として指定できることは明白であろう。さらに、2つよりも多い言語は混合入力テキスト文字列を形成することができる。

【0121】

<結論>

上の説明では、構造機能および／または方法論的動作に固有の言語を使用して、付属の請求項で定義されている本発明は説明した特定の機能または動作に限られるわけではない。むしろ、特定の機能および動作は、本発明を実装する実施例として開示されている。

【図面の簡単な説明】

【図1】

言語入力アーキテクチャを実装する言語固有のワードプロセッサを備えるコンピュータシステムのブロック図である。

【図2】

言語入力アーキテクチャの実施例のブロック図である。

【図3】

構文解析またはセグメント化して異なる音節群に分けたテキスト文字列および、テキスト文字列に誤りが含まれると仮定してそれらの音節を置き換えるのに使用することができる候補を説明しているブロック図である。

【図4】

言語入力アーキテクチャで実行される一般的変換を示す流れ図である。

【図5】

言語入力アーキテクチャで採用されている確率ベースのモデルをトレーニングするために使用されるトレーニングコンピュータのブロック図である。

【図6】

一トレーニング手法を説明する流れ図である。

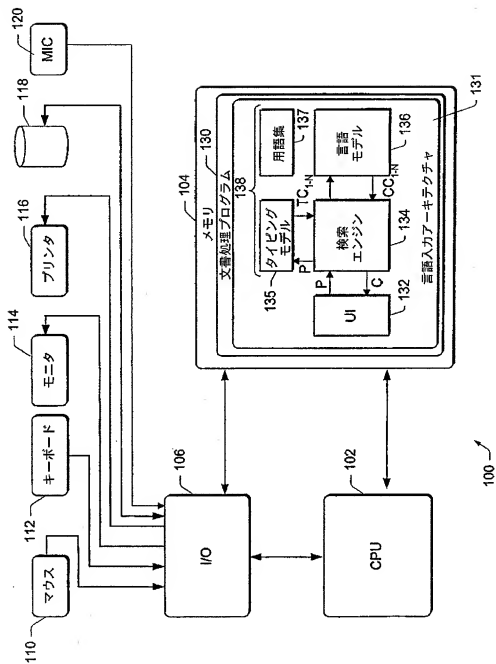
【図7】

複数のタイピングモデルを使用する、言語入力アーキテクチャの他の実施例のブロック図である。

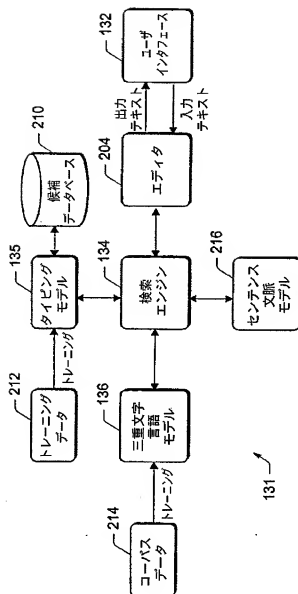
【図8】

他言語変換プロセスを説明する流れ図である。

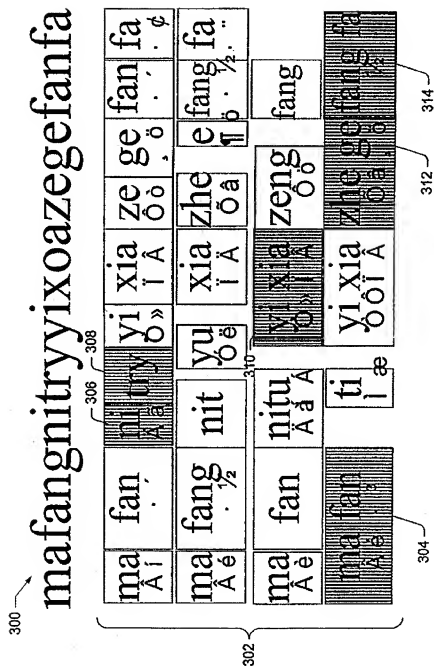
【図1】



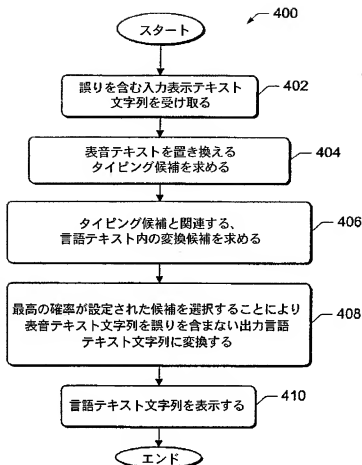
【図2】



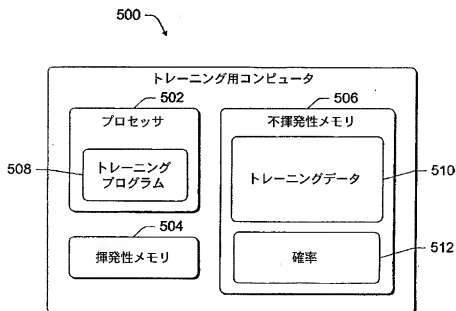
【图3】



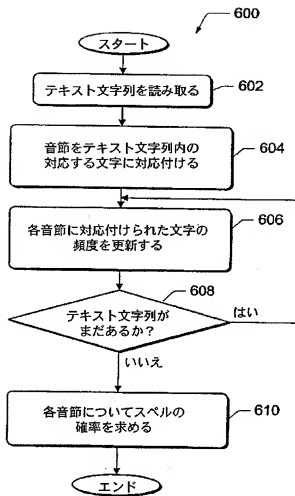
【図4】



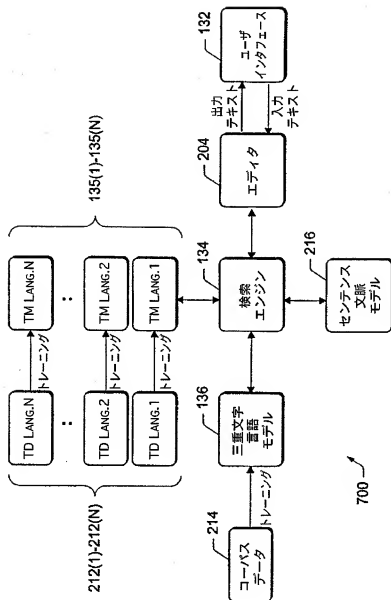
【図5】



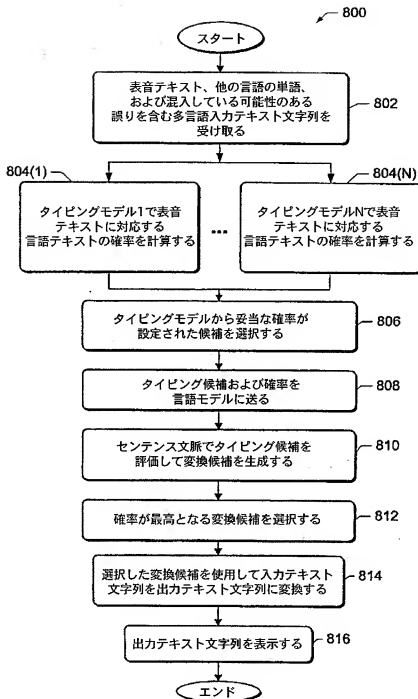
【図6】



【図7】



【図8】



【國際調查報告】

INTERNATIONAL SEARCH REPORT

International Application No.
PCT/US 00/28486

A. CLASSIFICATION OF SUBJECT MATTER
IPC 7 G06F17/27 G06F17/28

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 7 G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the International search (name of data base and, where practical, search terms used)

EPO-Internal, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indications, where appropriate, of the relevant passages	Relevant to claim No.
X	EP 0 555 545 A (IBM) 18 August 1993 (1993-08-18) abstract page 2, line 3 - line 6 page 2, line 36 - line 42 page 3, line 6 - line 9	1-53
A	US 5 893 133 A (CHEN CHENJUN JULIAN) 6 April 1999 (1999-04-06) abstract; figure 6 column 4 - column 5	6, 7, 16-18, 29, 35, 37, 39, 43-45

☒ Further documents are listed in the continuation of box C.

☒ Patent family members are listed in annex.

* Special categories of cited documents:

- *A* document defining the general state of the art which is not considered to be of particular relevance
- *E* earlier document but published on or after the international filing date
- *L* document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
- *O* document referring to an oral disclosure, use, exhibition or other means
- *P* document published prior to the international filing date but later than the priority date claimed

- *T* later documents published after the international filing date or priority date and not in conflict with the application but cited to underscore the principles or theory underlying the invention
- *X* document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
- *Y* document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
- *G* document member of the same patent family

Date of the actual completion of the international search

11 September 2001

Date of making of this international search report

19/09/2001

Name and mailing address of the ISA

European Patent Office, P.B. 5816 Patentbox 2
NL - 2209 HV Rijswijk
Tel (+31-70) 340-3340, Tx 31 651 epo nl,
Fax (+31-70) 340-3016

Authorised Officer

Triest, J

INTERNATIONAL SEARCH REPORT

International Application No.
PCT/US 00/28486

C. (Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT		
Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim no.
A	YUAN M ET AL: "A NEURAL NETWORK FOR DISAMBIGUATING PINYIN CHINESE INPUT" CALICO. ANNUAL SYMPOSIUM. PROCEEDINGS OF THE COMPUTER ASSISTED LANGUAGE INSTRUCTION CONSORTIUM. DISTANCE LEARNING. XX, XX, 14 March 1994 (1994-03-14), pages 239-243, XP001020457 the whole document	6-9,16, 17,37, 39,43,44
A	US 5 806 021 A (LIU FU-HUA ET AL) 8 September 1998 (1998-09-08) abstract	10,15
A	US 5 933 525 A (MAKHOUL JOHN I ET AL) 3 August 1999 (1999-08-03) abstract	1,31,51

INTERNATIONAL SEARCH REPORT

Information on patent family members

International Application No.

PCT/US 00/28486

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
EP 0555545 A	18-08-1993	US 5267345 A	30-11-1993
		JP 2069083 C	10-07-1996
		JP 5289692 A	05-11-1993
		JP 7107640 B	15-11-1995
US 5893133 A	06-04-1999	CN 1143769 A	26-02-1997
		JP 9128383 A	16-05-1997
		KR 259407 B	15-06-2000
		KR 268297 B	15-09-2000
		SG 55208 A	21-12-1998
		TW 411421 B	11-11-2000
		US 6073146 A	06-06-2000
US 5806021 A	08-09-1998	NONE	
US 5933525 A	03-08-1999	NONE	